



ANSYS ACT 19.1 Migration Notes

Copyright and Trademark Information

© 2018 ANSYS, Inc. Unauthorized use, distribution or duplication is prohibited.

ANSYS, ANSYS Workbench, AUTODYN, CFX, FLUENT and any and all ANSYS, Inc. brand, product, service and feature names, logos and slogans are registered trademarks or trademarks of ANSYS, Inc. or its subsidiaries located in the United States or other countries. ICEM CFD is a trademark used by ANSYS, Inc. under license. CFX is a trademark of Sony Corporation in Japan. All other brand, product, service and feature names or trademarks are the property of their respective owners. FLEXlm and FLEXnet are trademarks of Flexera Software LLC.

Disclaimer Notice

THIS ANSYS SOFTWARE PRODUCT AND PROGRAM DOCUMENTATION INCLUDE TRADE SECRETS AND ARE CONFIDENTIAL AND PROPRIETARY PRODUCTS OF ANSYS, INC., ITS SUBSIDIARIES, OR LICENSORS. The software products and documentation are furnished by ANSYS, Inc., its subsidiaries, or affiliates under a software license agreement that contains provisions concerning non-disclosure, copying, length and nature of use, compliance with exporting laws, warranties, disclaimers, limitations of liability, and remedies, and other provisions. The software products and documentation may be used, disclosed, transferred, or copied only in accordance with the terms and conditions of that software license agreement.

ANSYS, Inc. and ANSYS Europe, Ltd. are UL registered ISO 9001: 2008 companies.

U.S. Government Rights

For U.S. Government users, except as specifically granted by the ANSYS, Inc. software license agreement, the use, duplication, or disclosure by the United States Government is subject to restrictions stated in the ANSYS, Inc. software license agreement and FAR 12.212 (for non-DOD licenses).

Third-Party Software

See the legal information in the product help files for the complete Legal Notice for ANSYS proprietary software and third-party software. If you are unable to access the Legal Notice, contact ANSYS, Inc.

Published in the U.S.A.

Table of Contents

| | |
|--|---|
| ANSYS ACT 19.1 Migration Notes | 1 |
| Certain assemblies are no longer automatically imported into the script scope | 1 |
| New plug-ins for Workbench CAD import may affect entity count and topoID values | 1 |
| Removal of properties from class "TaskTemplate" in namespace "Ansys.ACT.Workbench.Automation.Workflows" | 1 |
| Deprecation of methods in namespace "Ansys.ACT.Automation.Mechanical" | 2 |
| Setting the X axis in a graph in Mechanical | 3 |
| Changes to Mechanical's Automation API methods for user interface transactions..... | 3 |
| Setting phase angle tabular data for a remote force | 3 |
| Modification to the property <code>ThreadAngle</code> for a contact region..... | 3 |
| Semantics change for comparing Mechanical tree objects..... | 3 |

ANSYS ACT 19.1 Migration Notes

As improvements are made to ACT APIs and the way that they display and transmit data, great efforts are taken to ensure that changes are backwards-compatible. For your convenience, this section lists API changes that might impact your existing extensions so that you can determine if any action is necessary before migrating them to ANSYS 19.1.

Certain assemblies are no longer automatically imported into the script scope

Certain assemblies, such as `Ans.Core.dll` and `Ans.ProjectSchematic.dll`, are no longer automatically imported into the script scope at 19.1. If you receive an error such as `NameError: name 'Ansys' is not defined`, add the assembly reference to the `clr` as shown in this example:

```
import clr
clr.AddReference("Ans.ProjectSchematic")
import Ansys.ProjectSchematic
```

New plug-ins for Workbench CAD import may affect entity count and topID values

For the IGES and STEP file formats, the Workbench CAD import uses new plug-ins at 19.1. As a result, entity count and topID values may not match past releases. Review and, if necessary, update your ACT scripts to use the new entity count and topID values generated through the import process.

Removal of properties from class "TaskTemplate" in namespace "Ansys.ACT.Workbench.Automation.Workflows"

Two properties have been removed from the public class `TaskTemplate` in the namespace `Ansys.ACT.Workbench.Automation.Workflows`:

- `IsPartialShareOnly`
- `DuplicateUserDataCommmandName`

Because these two properties no longer exist in the ANSYS Workbench framework, you can no longer use them in ACT extensions for customizing Workbench workflows.

Deprecation of methods in namespace

"Ansys.ACT.Automation.Mechanical"

Several methods in the namespace `Ansys.ACT.Automation.Mechanical` are deprecated in 19.1. You can download and extract [scripts](#) for helping you to migrate your extensions. Tentative plans exist for removal of the deprecated methods in ANSYS 20.0.

Below, primary bullets list the deprecated methods. Under each primary bullet, child bullets list the new replacement methods. `IExtension` is `ExtAPI.ExtensionManager.CurrentExtension`.

- `analysis.CreateLoadObject(string objName)`
 - `analysis.CreateLoadObject(string objName, string extNameOrGUID)`
 - `analysis.CreateLoadObject(string objName, IExtension ext)`
- `analysis.CreateResultObject(string objName)`
 - `analysis.CreateResultObject(string objName, string extNameOrGUID)`
 - `analysis.CreateResultObject(string objName, IExtension ext)`
- `analysis.CreatePostObject(string objName)`
 - `analysis.CreatePostObject(string objName, string extNameOrGUID)`
 - `analysis.CreatePostObject(string objName, IExtension ext)`
- `analysis.LoadObjects`
 - `analysis.GetLoadObjects(string extNameOrGUID)`
 - `analysis.GetLoadObjects(IExtension ext)`
- `analysis.ResultObjects`
 - `analysis.GetResultObjects(string extNameOrGUID)`
 - `analysis.GetResultObjects(IExtension ext)`
- `analysis.UserObjects`
 - `analysis.GetUserObjects(string extNameOrGUID)`
 - `analysis.GetUserObjects(IExtension ext)`
- `dataModel.CreateObject(string objName)`
 - `dataModel.CreateObject(string objName, string extNameOrGUID)`
 - `dataModel.CreateObject(string objName, IExtension ext)`
- `dataModel.UserObjects`
 - `dataModel.GetUserObjects(string extNameOrGUID)`
 - `dataModel.GetUserObjects(IExtension ext)`

Setting the X axis in a graph in Mechanical

In the Mechanical API, the exposed property `GraphControlsXAxis` was changed from an `int` type to an `enum` type so that you can now easily select the field that you want to graph from a list.

Changes to Mechanical's Automation API methods for user interface transactions

In 19.0, the method `with Transaction()` replaced the deprecated method `with ExtAPI.DataModel.Tree.Suspend()`. When using Mechanical's Automation API to create or modify many objects in the Mechanical tree, the method `with Transaction()` allows you to define the scope of a user interface transaction so that only one refresh is performed at the end of the transaction. By eliminating refreshes after every method call, this method significantly improves scripting performance. Sample code follows:

```
with Transaction():
    contacts = ExtAPI.DataModel.Project.Model.Connections.Children[0]
    for contact in contacts.Children:
        contact.ContactType = ContactType.NoSeparation
```

Setting phase angle tabular data for a remote force

In 19.0, an issue in the Mechanical API with the exposed property `PhaseAngle` was fixed so that you can set phase angle tabular data for a remote force. As a part of this fix, the return type was changed from `quantity` to `field`.

Modification to the property `ThreadAngle` for a contact region

In 19.0, the return type for the property `ThreadAngle` was changed from `double` to `quantity` in the Mechanical API. With this change, the property `ThreadAngle` now behaves like other angle properties.

Semantics change for comparing Mechanical tree objects

Due to a 19.0 change in the Automation API, to compare Mechanical tree objects, use the method `Equals()` or `"=="` (in IronPython only) rather than using the keyword `is`.

ANSYS ACT 19.1 Migration Notes

In the sample code that follows, two variables reference the same project in the Mechanical tree:

```
project = ExtAPI.DataModel.Project  
same_project = ExtAPI.DataModel.Project
```

The operation `project is same_project` will return `False`. Instead, use `project.Equals(same_project)` or `project == same_project`.

Note: ACT has superseded the ANSYS Workbench Software Development Kit (SDK) and External Connection Add-In as the best-in-class tool set for customizing ANSYS products. Support for the SDK and External Connection Add-in has ended as of 19.0. If you have used these deprecated tools for Workbench customizations, see the [ANSYS SDK and External Connection Add-in Migration Guide](#) on the [ACT Resources page](#) of the ANSYS customer site for migration information.